Towards enforcement of purpose for privacy policy in distributed healthcare

Rath, Thavy Mony Annanda; Colin, Jean-Noël

Published in:
2013 IEEE 10th Consumer Communications and Networking Conference, CCNC 2013

DOI:
10.1109/CCNC.2013.6488578

Publication date:
2013

Document Version
Early version, also known as pre-print

Towards Enforcement of Purpose for Privacy Policy in Distributed Healthcare

Annanda Thavymony Rath
PReCISE Research Center, Faculty of Computer Science
University of Namur, Belgium
Email: rta@info.fundp.ac.be

Jean-Noël Colin
PReCISE Research Center, Faculty of Computer Science
University of Namur, Belgium
Email: jean-noel.colin@fundp.ac.be

***Abstract*—Purpose of access is one of the core concepts in privacy which considers the data user's intent as a factor in making access control decisions and enforcement of purpose is required to ensure that data is used as what it intends for. In general, the enforcement of purpose is a complicated task. The main difficulty is how to identify the purpose of an agent when it requests to perform an action. In this paper, we discuss the design issue of purpose enforcement based on our proposed (defined) enforcement structure: pre-enforcement, ongoing-enforcement, and post-enforcement. We also propose an enforcement solution for usage control designed for distributed healthcare information system, particularly, the pre-enforcement of purpose (the validation of claimed purpose at the initial state before data is granted access).**

***Keywords*-purpose enforcement; distributed healthcare; security; privacy.**

I. Introduction

In dictionary, "purpose" is defined as "the object towards which one strives or for which something exists; an aim or a goal". However, by observing how purpose is used in the natural language reveals that purposes often refer to an or a set of abstract actions. For example, accessing patient's health record for the purpose of treatment, research, insurance, etc. all of which are names of some abstract actions. Jafari et al [5] classified "purpose" in two types: purpose as high-level action and purpose as future action.

Purpose as a high-level action refers to a more abstract, or semantically higher-level action in a plan. Thus, doing something for some purpose, actually means doing it as a part, or a sub-action, for that higher-level action. For example, when Bob checks some patient's blood pressure record for the purpose of heart surgery, it means that checking the blood pressure is a part of a more complex and abstract action of heart surgery. Similarly, surgery is performed for the purpose of treatment, it is because the high-level action of medical treatment includes surgery as a part.

Purpose as a future action is used to indicate that an action is performed as a prerequisite of another action in future. For example, when Bob withdraws money from a bank account for the purpose of paying the bills, it means the former action is done as a prerequisite to performing the latter.

To Jafari et al [5], the enforcement of purpose means to verify that those abstract actions exist and they are valid before data is released to requester. In some contexts, they need also to be valid during the usage of data. Identifying purpose of action is the main difficulty in purpose enforcement. Some common proposed mechanisms for purpose management and enforcement are self-declaration in which the agent explicitly announces the purpose of data access [3] and role-based enforcement [7] in which the purpose is identified based on the agent's role in the system. The first method obviously cannot stop a malicious agent from claiming false purposes. This is because anyone can claim any purpose of access, without the proper system to validate claimed purpose, this method can not be used in data processing environment like distributed healthcare [1][2]. The second method has been criticized to be inefficient in capturing purpose of an action since roles and purposes are not always aligned and members of the same organizational role may practice different purposes in their actions. Therefore, identifying the purpose of action or verifying the claimed purpose remains an open question.

This paper addresses two main issues: (a) propose the purpose enforcement structure. (b) propose the design of purpose validation for the three validation phases (pre-, ongoing-, and post- enforcement) for distributed healthcare.

The rest of the paper is organized as follows. Section II presents the motivation and related work. Section III talks about the purpose enforcement structure. We present a purpose enforcement model in Section IV and a prototype of the proposed model in Section V. Section VI is the conclusion and future work.

II. Motivation and related work

Purpose is raised and argued in many literatures as an important entity used to control access to sensitive private data. Byun et al [3][9] proposed a purpose-based access control model of complex data for privacy protection, a model that relies on the well-known RBAC [4] access control model as well as the notion of conditional role which is based on the concept of role attribute and system attribute. In their paper, they provided also a general purpose tree applied in complex data management system and a solution to address the problem of how to determine the purpose for which certain data are accessed by a given user.

Jafari et al [5] defined a semantic model for purpose, based on which purpose-based privacy policies can be expressed and enforced in a business system. The model is based on the

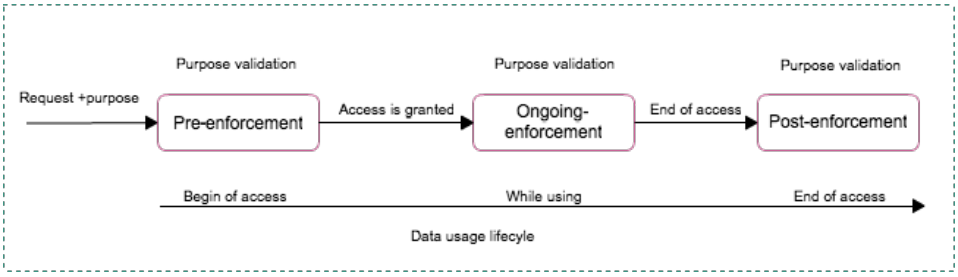

Fig. 1. Purpose enforcement structure

intuition that the purpose of an action is determined by its situation among other inter-related actions. Actions and their relationships can be modeled in the form of an action graph. A modal logic and model checking algorithm are developed for formal expression of purpose-based policies and for verifying whether a particular system complies with them.

Concerning enforcement, Katt et al [8] proposed the extension of $UCON_{ABC}$ [10] with continuous control usage sessions for expressing the ongoing-check obligation. They also proposed the general, continuity-enhanced policy enforcement engine for usage control applied particularly to obligation. After the thorough study on the work of Katt et al, we found that the model can be extended and used to enforce the ongoing-enforcement of purpose. We will discuss it in detail in the following section.

Jafari et al [6] proposed an approach to enforce purpose in access control systems that uses workflows. They proposed to encode purposes as properties of workflows used by organizations. However, the proposed model does not work with "purpose" that does not have a natural interpretation in terms of workflows, particularly, more abstract purposes.

III. Purpose enforcement structure

"Purpose" has been considered in major privacy legislations [1] where the processing of sensitive private data is bounded to the specific purpose and the excessive use of them are prohibited. With this regard, in any processing environment dealing with such data requires great attention to make sure that system can provide adequate data processing security aligning with privacy legislation. This leads to the necessity of the effective management of purpose binding of data (including the recognition of purpose binding data) and enforcement.

Observing how data is processed in the real world reveals that there are three crucial states that need to be considered for the enforcement of purpose: before access is granted (initial state), during the usage of data, and after using it. However, the three enforcement states require different mechanisms to handle them. We term the three enforcement states as pre-enforcement, ongoing-enforcement, and post-enforcement respectively. Figure 1 shows the purpose enforcement structure with the three enforcement states.

[1]Privacy legislations: the European 95/46/EC Directive, U.S Privacy Act (1974), and Canada's Federal Privacy Act (1983)

- Pre-enforcement refers to a mechanism allowing the system to validate the purpose before granting access to data. At this stage, the user's request in which the purpose of access is mentioned is validated by the system. If the system finds that the claimed-purpose is not valid, it rejects request immediately without going further into the detail evaluation of the usage policy. For example, in emergency case, if doctor declares "emergency purpose" in order to bypass the rule to access patient's record and if system can not prove the existence of "emergency situation", then the request is rejected.
- Ongoing-enforcement refers to a mechanism allowing the system to continuously control purpose of usage during the usage period. It checks if the actions performed and the requesting actions comply with the claimed purpose. During the usage, system periodically triggers the re-evaluation of purpose, this intends to check if the purpose of access is still valid given the change of time. Ongoing-enforcement can be called "controlling and guiding method" because it acts as a controller and also a guide for user. It tells the user which action is allowed for which purpose.
- Post-enforcement refers to a mechanism allowing the system to validate the processing of data and to identify if the usage of data was in line with the claimed purpose or otherwise. This enforcement is done after the usage of data. This mechanism can also be called a pro-active enforcement, it does not provide ongoing control of the data usage, instead it provides a way to prove the correctness of the data usage by means of the log information. Auditing mechanisms are required to analyze the log-information and to reconstruct the execution process in order to find out if a violation happened or not.

With the above consideration, we see that to ensure the correctness of data usage, the purpose in those three states must be maintained. To support this enforcement structure, we propose the enforcement model as presented in next section.

IV. Purpose enforcement model

In this section, we present in detail the purposed-based usage enforcement model applied in distributed healthcare information system. The enforcement model focuses on the system architecture and functional modules to illustrate how the enforcement can be achieved. The proposed architecture

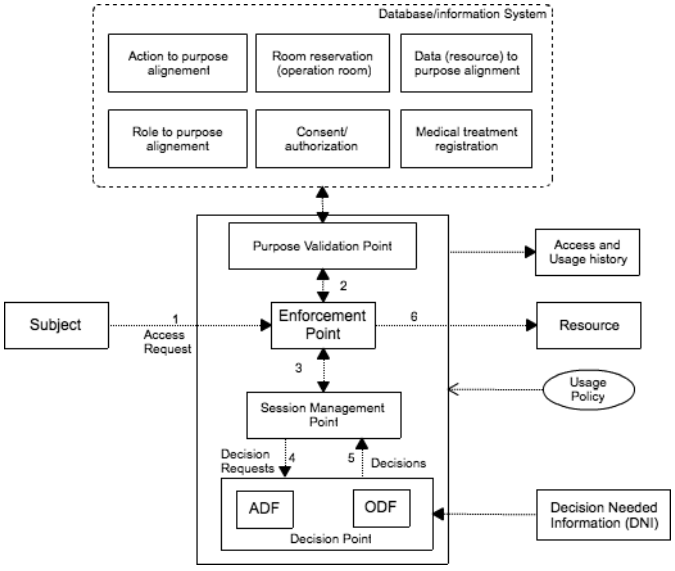

Fig. 2. Purpose-based usage control enforcement model

considers all the UCON core model [11] and it is inspired partly from the work of Katt et al [8]. As illustrated in Figure 2, the model consists of four core components: Purpose Validation Point (PVP), Enforcement Point (EP), Decision Point (DP), and Session Management Point (SMP) with other supplementary modules like Usage Policy, Access/Usage History, Decision Needed Information (DNI) or Information Point (IP), and Information system for purpose validation.

1) DP is responsible for making the access decisions during a usage control session. It consists of two decision-making components:
   - Attribute Decision Function (ADF) handles the attribute-based access decision during a usage session. Attributes can be either subject, object, or environment attributes (e.g., the subject's identification). The information required by ADF is retrieved from DNI.
   - Obligation Decision Function (ODF) makes the decision whether a specific obligation has been fulfilled. The DP checks the fulfillment of an obligation by transforming it into an ordered sequence of system actions, which should be defined for all obligations (e.g., notification, the system may requires to check if an email address is valid, and if it is valid, the further check is if the notification message is successfully sent out through the provided mail address). During the obligation fulfillment check process, in case, the DP requires more information needed in obligation evaluation process, it contacts DNI.
2) PVP makes the decision whether a purpose is valid or not. Whenever there is a request, PVP checks the request based on the claimed purpose. To validate the usage purpose, PVP contacts the purpose information system consisting of different modules responsible for providing the information concerning different types of purpose (Figure 2). Below are the details of those components.

- "Role to purpose alignment" provides the information concerning the alignment between the requester's role and the purpose of access. For example, a requester in role of "cardiologist" may be aligned to the "heart surgery purpose". This information can be used for the pre-enforcement of purpose, particularly, at the initial state of access. However using the role to purpose alignment alone may not be an effective solution to the problem as roles and purposes are not always aligned. Thus, this information needs to be used in conjunction with other information from other modules presented below.
- "Action to purpose alignment" provides the information concerning the alignment between the actions on object and the purpose. For example, action "transfer" is aligned to the "emergency purpose". However, like "role to purpose", "action to purpose" can be used only as the complement to other modules for purpose enforcement.
- "Data to purpose alignment" provides the information concerning the alignment between type of object (resource or data) and purpose. For example, data concerning surgery may be aligned to the request for "surgery purpose".
- "Medical treatment registration", in general, patient needs to register for the medical check, the registration information can be used to prove if the purpose claimed by the requester is inline with the treatment of the patient. For instance, if the patient registered for general normal medical check, the requester's (e.g, doctor or physician) claim of purpose as "emergency" is not valid.
- "Room reservation (operation room or emergency room)" provides the information concerning the room reservation for each operation. This module is designed as the source of information in case of emergency situation. For example, in case of emergency situation, the access rule on data may be bypassed; hence, operation room reservation can be the source of information to validate the claimed-purpose.
- "Consent/authorization" provides the information about who is particularly authorized for which purposes. This module is designed to be used for two purpose categories: Administrative and the others (including research or health insurance). This module is administrated by the trusted entity that has the authority to align a particular user or a group of users to the particular purposes. For example, user "Bob" can access patient's record for "research" purpose.

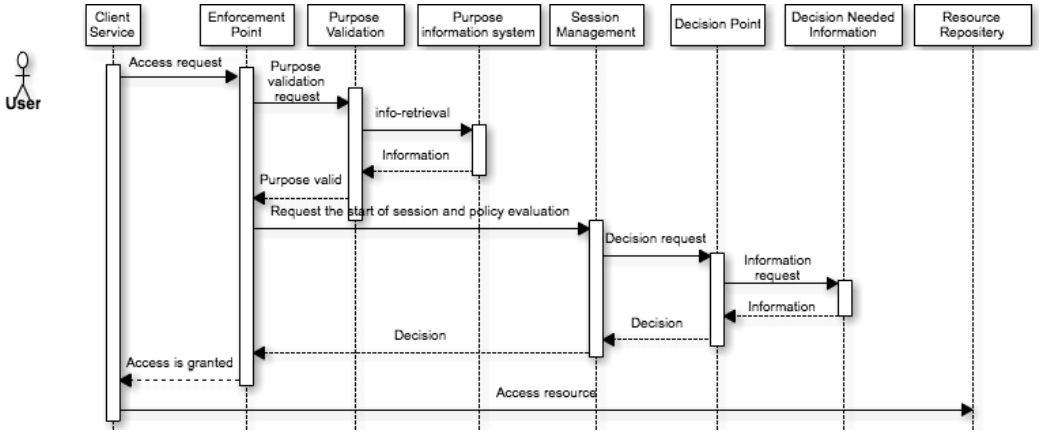

Fig. 3. Functioning of enforcement system (positive response)

3) EP handles the request from the subject and forwards it to purpose validation point and then to decision point through session management point for further policy evaluation. If the usage request is granted by DP, then EP allows the subject to access resource, else, a denying message is sent to subject.
4) SMP is the module that manages individual usage session. This includes requesting required decision(s) from concerning modules (ADF and ODF) in each state during the usage session. When a usage request is received, EP forwards it to SMP. SMP sends corresponding decision requests to DP. Then DP launches a checking process for all the concerned modules such as ADF and ODF. If all the requirements are fulfilled, DP sends granting message to SMP and SMP forwards it to EP. In case a negative decision is received, SMP sends a denial response to EP. In the accessing state, the SMP monitors continuously related subject, object, and environment attributes, as well as any further actions requested by the subject. According to the decision received from DP, SMP either revokes the ongoing access by sending a revoked response to EP, or keeps permitting access.

Remark: in general, "purpose" is included in the policy and the enforcement of it can be done after the decision by Decision Point (Figure 2). However, in our proposed model, we consider purpose validation as a separate module. This module examines the purpose of access at the early state, before even the usage session starts. The logic behind this is that we intend to maximize the performance of the system. If we let everything be processed at the DP, it would take considerable time in case the purpose of access is not valid because PDP may also evaluate other parameters in policy. Thus, we adopt the multi-tier control approach by having purpose validation first and then the detail UCON policy evaluation second [11] .

V. Model implementation

To show the functionality of our proposed model, we designed a concrete usage control enforcement engine as presented in Figure 4. We then developed and tested a proof-of-concept prototype in Java program. Furthermore, in order to utilize and facilitate the existing standards and frameworks in the area of access control, XACML "enterprise-java-xacml"[2] is used in this prototype as a core policy evaluation engine. Given that the usage enforcement should be done in a remote client platform, we have no control over the remote system. That requires trust establishment between the service provider and the remote client before any data is released. While the trust issue goes beyond the scope of this paper, we assume that the remote client is trusted before the usage control policy enforcement takes place.

Concerning the meta model and the state transition model applied to Figure 2 (e.g, a state transition model for SMP), we adopted the model proposed by Katt et al [8] and Zhang et al [11] respectively.

A. Prototype

In our eHealth scenario, the doctor requests patient's health record from the Healthcare Information System (HIS). After authenticating and authorizing the doctor based on his/her role and a purpose of access, the HIS releases the record and a usage policy in one encrypted package. The encrypted package can reside on the doctor device for a specific period of time during which doctor can re-access/re-use it. The enforcement component, which is integrated into the document reader (at client side), checks the integrity of the package and extracts the usage control policy and the patient's record. Figure 4 shows the architecture of our usage control enforcement engine. The implementation is based on XACML engine and the work of Katt et al [8]. In general the engine consists of the following components:

- PEP acts as single entry point to protected resources and performs an access control. It receives the usage requests from requester, and first makes a purpose validation request (PVQ) and consequently receives the response (PVR) from PVP. After receiving the positive response

[2]http://code.google.com/p/enterprise-java-xacml/

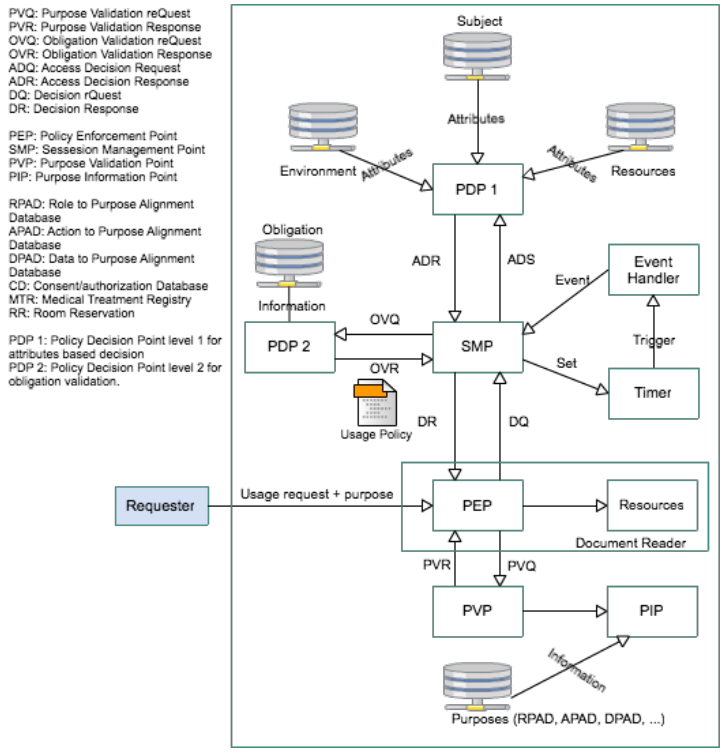

Fig. 4. Implementation architecture

from PVP, it makes a usage decision request (DQ) to PDP 1 and PDP 2 through SMP and gets the decision response (DR) either positive or negative. Then SMP forwards the response to PEP. PEP enforces the authorization decisions it receives by either allowing access or denying it.

- PVP acts as a validation point for purpose, which needs to be verified before further validation of the usage policy by PDP 1 and PDP 2. If PVP provides a negative response, the process ends and no further evaluation of usage policy. In case of positive response from PVP, a further decision request is sent to PDP 1 and PDP 2 through SMP for further usage policy evaluation.
- PIP provides all the necessary information to PVP during the validation state. The information provided to PVP comes from different modules (Figure 2). Those modules are responsible for providing information for different types of purpose. For example, in case of emergency purpose, the role to purpose alignment and medical treatment registration modules are the sources of information.
- SMP is the dynamic part of the whole engine and captures the continuity behavior of the access control system. Furthermore, it manages the functions of other elements of the architecture and ensures the transitions from one state to another according to the extended UCON states transition [8][11]. Initially, the SMP is in the initial state. Upon receiving a DQ request from PEP, the initial-request transition is triggered and SMP moves to the requestCheck state[3]. In this state SMP requests authorization and obligation decisions from the PDP 1 and the PDP 2, respectively. It is worth noting that the change from one state to another is defined by the state change rule [8], which is executed by the event handler.
- Event handler handles the events that trigger transitions from one state to another. It listens to the events and sends the trigger actions to SMP when state change is about to occur.
- Timer can be set by the SMP through the event handler. Timer can be used for supporting obligations with deadlines and obligation re-evaluation.
- PDP 1 refers to ADF function in our enforcement model and is represented as a XACML PDP. It is the component that evaluates attribute-related constraints (authorizations and conditions) and renders decisions to SMP.
- PDP 2 refers to ODF function of our enforcement model. It receives an obligation request from the SMP and checks whether the obligation has been fulfilled. As in this paper, we focus on purpose enforcement, we provide the detail of its design in the following section while more detail on the design of ODF can be found in [8].

B. Design of Purpose Validation Point

Based on our proposed model, claimed purpose of access by requester needs to be validated by the system based on the purpose alignment policy (which expresses how "pur-

[3]requestCheck is a state at which the system re-evaluates a usage policy.

pose" should be evaluated in XML format). The purpose of access is classified into two types: normal and emergency. In normal case, all the six information (Figure 2, ranging from "action to purpose alignment" to "medical treatment registration", most importantly "consent/authorization") modules need to be checked while in case of emergency access, consent/authorization module is ignored. This means in case of emergency, unconsented doctor can also access. After getting information from PIP, PVP performs the evaluation process and sends the positive or negative response to PEP. Positive response means the purpose is valid while negative signifies otherwise. In case of positive response, PEP goes to further step by initiating the decision request to PDP 1 and PDP 2 through SMP. It is worth noting that the PVP acts upon every usage request.

C. Design of the request structure

The request must contain: "User", it can be identified by their name or identity. "Role", it is a role of user in a particular institution. For example, cardiologist can be considered as user role. It is worth noting that in our implementation, we assume that user to role validation is done by other module and we do not address it here. "Action on resource" is a requested action performing on resource. For example, transfer, copy, or read. "Resource" is a targeted object by requester. "Purpose of access", it is a claimed purpose by user, this is an important element in the request, user must declare their access purpose when initiating request. It is also worth noting that we adopt the XACML's request structure in our implementation where the purpose of access is encoded in the environment attribute specified in standard XACML request.

D. Implementation and testing

To implement our proposed system architecture, we built five components in Java that form a usage control application. The first component is the document reader, which is responsible for processing the requested resource in the secure way. The second component is PVP, which is connected to the purpose information point. In our implementation the six components of the purpose information point (role to purpose alignment, ... refer to Figure 2) are encoded in XML format. The third component is the event handler, we use Java Timer to set a time for triggering the event during the usage session. The fourth component is the ADF module, which is based on the enterprise-java-xacml used for usage policy evaluation. The fifth component is the ODF that is built based on the work of Katt et al [8]. To test our application, we created different type of policies for both normal and emergency situation. Based on the result of the test, we noticed that in case of an invalid purpose, the request response time is small. This is because according to our proposed architecture, the system bypasses the detail evaluation of policy (this may involve the evaluation of other attributes) and the decision is cut short by PVP.

VI. Conclusion and future work

In this paper, we addressed the issue of purpose enforcement in access and usage control, applied to eHealth domain as an illustration. We proposed a classification for enforcement mechanisms, based on the moment they happen in the access timeline and defined pre-, ongoing-, and post-enforcement. Building on this classification, we proposed an original model for purpose enforcement, as well as a system architecture that introduced some generic components that contribute to the enforcement of access purpose. A prototype of the model has been developed as a first step into validation.

In this paper, we only dealt with the pre- and ongoing-enforcement cases. In the future, we plan to extend our model to handle the post-enforcement case. We also aim at extending the purpose enforcement engine to adopt a probabilistic behavior. The intuition behind this is that in most situations, it is not possible to get a 100 percent sure validation of the purpose of access, or some validation can only be performed after access has been granted, thus the need for a probabilistic approach. Finally, we worked mainly with (variants of) the RBAC and UCON model, and would like to extend our model to other access control models.

References

[1] Annanda Thavymony RATH and Jean-Noël Colin. *Patient privacy preservation: P-RBAC vs OrBAC in patient controlled records type of centralized healthcare information system, case study of walloon healthcare network, Belgium*. The Fourth International Conference on eHealth, Telemedicine, and Social Medicine eTELEMED 2012 4 111-118, Spain: Valentia, 2012.

[2] Annanda Thavymony RATH and Jean-Noël Colin. *Analogue attacks in e-health: Issues and solutions*. CeHPSA - 2012 : 2nd IEEE International Workshop on Consumer eHealth Platforms, Services and Applications (CeHPSA), USA: Las Vegas, 2012(accepted but unpublished).

[3] Byun Ji-won, Bertino elisa, and Li Ninghui. *Purpose based access control of complex data for privacy protection*. In Proceedings of the tenth ACM symposium on Access control models and technologies (New York, NY, USA, 2005), SACMAT 05, ACM, pp. 102-110., USA: New York.

[4] D.F.Ferraiolo, R. Sandhu, S. Gavrila, D.r.kuhn, and R.Chandramouli. *Proposed NIST Standard for Role- Based Access Control*. In ACM Transactions on Information and System Security (August 2001), pp. 4(3):222-274.

[5] Jafari Mohammad, Fong Philip, Safavi-Naini Rei-Haneh, Barker Ken, and Sheppard Nicholas Paul. *Towards defining semantic foundations for purpose-based privacy policies*. In Proceedings of the first ACM conference on Data and application security and privacy (San Antonio, TX, USA, 2011), CODASPY 11, ACM, pp. 213-224.

[6] Jafari Mohammad, Safavi-naini Rei-Haneh, and Sheppard Nicholas paul. *Enforcing purpose of use via workflows*. In Proceedings of the 8th ACM workshop on Privacy in the electronic society (New York, NY, USA, 2009), WPES 09, ACM, pp. 113-116.

[7] Jawad Mohamed, Alvarado Patricia Serrano, and Valduriez Patrick. *Design of priserv, a privacy service for DHTS*. In Proceedings of the 2008 international workshop on Privacy and anonymity in information society (New York, NY, USA, 2008), PAIS 08, ACM, pp. 21-25.

[8] Katt Basel, Zhang Xinwen, Breu Ruth, Hafner Michael, and Seifert Jean-Pierre. *A general obligation model and continuity enhanced policy enforcement engine for usage control*. In Proceedings of the 13th ACM symposium on Access control models and technologies (New York, NY, USA, 2008), SACMAT 08, ACM, pp. 123-132.

[9] Ni.Qun, Bertino Elisa, Lobo Jorge, Brodie Carolyn, Clare-marie Karat, and Trombeta Alberto. *Privacy-aware Role-Based Access Control*. ACM Transaction Information and System Security 13 (July 2010), 24:1-24:31.

[10] Park Jaehong and Sandhu Ravi. *Towards usage control models: beyond traditional access control*. In Proceedings of the seventh ACM symposium on Access control models and technologies (New York, NY, USA, 2002), SACMAT 02, ACM, pp. 57-64.

[11] Zhang Xinwen, Parisi-presicce Francesco, Sandhu Ravi, and Park Jaehong. *Formal model and policy specification of usage control*. ACM Trans. Inf. Syst. Secur. 8 (November 2005), 351-387.